

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY-
DHARWAD

INTRODUCTION TO ALGORITHMS

EC351

ASSIGNMENT 1

FIBONACCI SERIES

Submitted by:

TEAM 3

SEDHURAMAN V 18BEC041

VARUN KUMAR 18BEC048

T K DERINVISHAL 18BEC047

Submitted To:

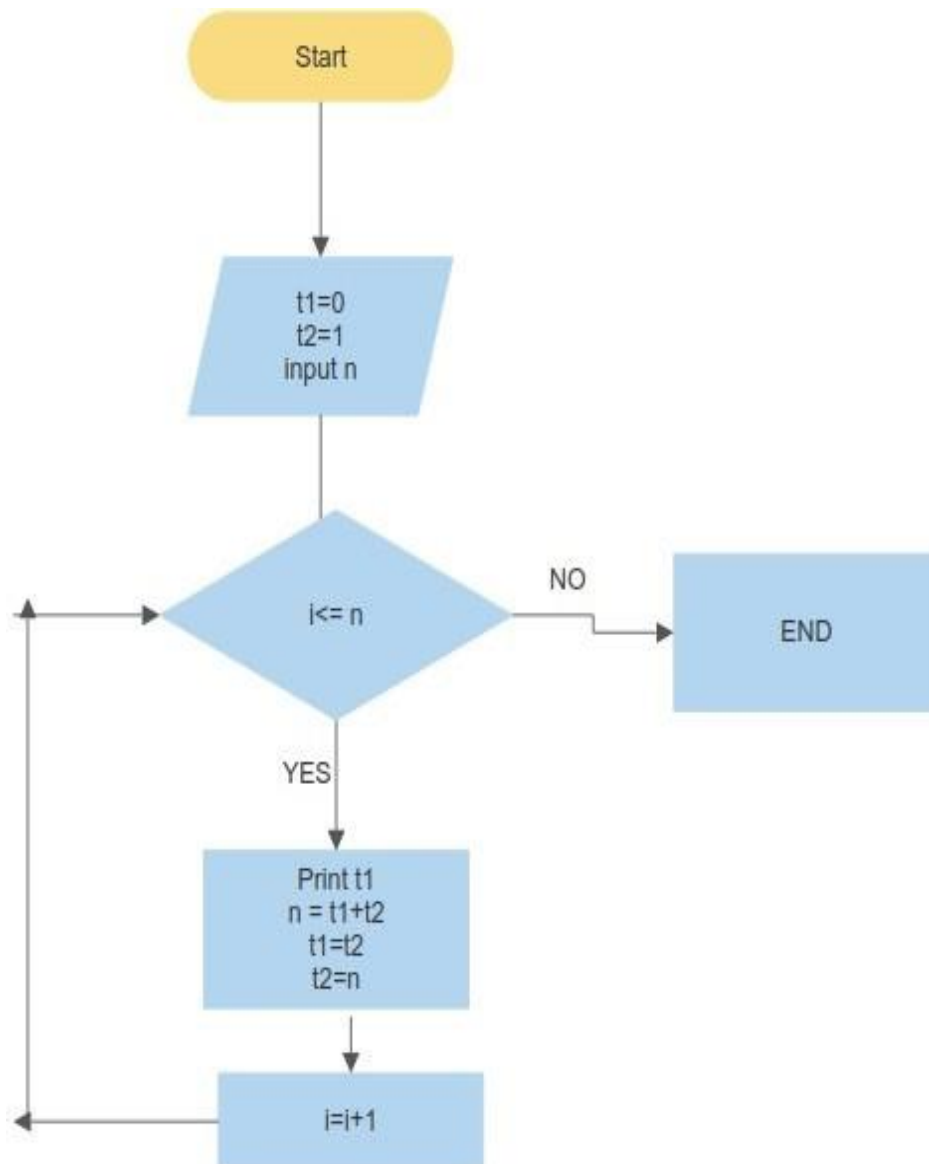
DR. UMA S

04 Sept 2020

QUESTION: Consider the following Fibonacci series and solve the following conditions

$\text{fib}(n) = \text{fib}(0), \text{fib}(1), \text{fib}(2), \dots, \text{fib}(n)$
where $\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2)$

- **Algorithm chart**



Iterative Code

```
#include <stdio.h>
int count = 0;

int main() {
    int i, n, t1 = 0, t2 = 1, nextTerm;
    printf("Enter the number of terms:");
    scanf("%d", &n);
    printf("Fibonacci Series:");

    for (i = 1; i <= n; ++i){
        printf("%d, ", t1);
        nextTerm = t1 + t2;
        t1 = t2;
        t2 = nextTerm;
        count++;
    }
    printf("\nVariable used %d times\n", count);
    return 0;
}
```

Recursive Code

```
#include <stdio.h>
int count = 0;
int x, y, n;

int fib(int n){
    count++;
    if(n == 0 || n == 1)
        return n;
    else
        return fib(n-1) + fib(n-2);
}

int main()
{
    printf("Enter the number");
    scanf("%d", &n);
    printf("%d\n", fib(n));
    printf("Variable used %d times\n", count);
}
```

- OBSERVATIONS

- For the **iterative approach**, the same amount of space required for $\text{fib}(5)$ and $\text{fib}(500)$, i.e. as N changes the space/memory used remains the same. It's space complexity is $O(1)$ or constant.
- For the iterative approach, the different amount of space required for $\text{fib}(5)$ and $\text{fib}(500)$, the maximum depth is proportional to the N , hence for Fibonacci recursive, the space complexity is **$O(N)$** .
- **The best case scenario** is iterative approach as the space complexity is $O(1)$.
- **Worst Case scenario** is recursive approach where space complexity is $O(N)$